PROBABILITY DISTANCE BASED COMPRESSION OF HIDDEN MARKOV MODELS

HAO WU[†‡] AND FRANK NOÉ[†§]

Abstract. Large-scale stochastic models are relevant in many different fields such as computational biology, finance, social sciences, communication and traffic networks. In order to both efficiently simulate and analyze such models and to understand the essential properties of the system, it is desirable to have model reduction techniques that much reduce the dimensionality of the model while at the same time preserving the system's essential dynamical properties. In this paper, a general model reduction technique for the class of discrete space and time Hidden Markov Models is presented, thereby also including the more special class discrete Markov Chains. The method is illustrated on some model applications.

Key words. hidden Markov model, total variation distance, compression, fuzzy partition

AMS subject classifications. 15B51, 60E15, 60J22, 90C26

1. Introduction. Large-scale stochastic models arise in many quantitative sciences, such as biophysics and physical chemistry [27, 28, 30], computational biology [36, 37], computational finance [33, 16], web navigation modeling [47], traffic modeling [26], and many others. In many cases of practical interest, the state space of models capturing the dynamics of the system at a faithful level of resolution is so large that simulating or analyzing them is computationally challenging and understanding their essential dynamical properties is a difficult task requiring significant expertise. Therefore, it is essential to have automatic methods that are able to compute models with a much reduced size while at the same time preserving the essential dynamical properties.

Most research in stochastic model compression has focused on Markov chains (MCs) because of their wide applicability and also because there is a well developed theory for MCs. The central idea of MC compression is to aggregate states which have similar dynamic characteristics. A well-known approach is based on the nearly completely decomposable Markov chain (NCDMC) model, which can be characterized by a transition matrix $A \in R^{N\times N}$ that takes the form

$$A = G + \epsilon H$$

where

$$G = \begin{bmatrix} G_{11} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & G_{22} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & G_{nn} \end{bmatrix}$$

$G_{ii} \in \mathbb{R}^{N_i\times N_i}$, $\sum_{i=1}^{n} N_i = N$ and ϵ is a small positive parameter. For systems of this type, the states can be clustered into n groups such that there is strong interaction within each group and weak interaction among the groups. Here, the clusters correspond to metastable states of the system. It was demonstrated for many applications,

[†]Mathematics Institute, Department of Mathematics and Computer Science, Free University of Berlin, Arnimallee 6, 14195 Berlin, Germany (hwu@zedat.fu-berlin.de, frank.noe@fu-berlin.de).

[‡]Supported by DFG grant NO 825/1 and SFB 449.

[§]Supported by DFG research center Matheon.

such as parameter estimation [22], solution of steady state probability distribution [2] and optimal control [1], that NCDMC based aggregated MCs can well approximate the behavior of the original model if ϵ is sufficient small. Several clustering approaches have been proposed to find the nearly completely decomposable structure in general transition matrices, such as lumping of rapidly-interconverting states [3, 43] and the Perron-cluster cluster analysis (PCCA) algorithm [38], which was later extended to include fuzzy clustering of MC states [44, 10]. Another widely used compression method was developed by Spears [41], which aggregates states having similar transition behaviors with respect to the remaining state space into a new state. For example, states i and j can be combined if the probabilities of transitions into states i, j and from states i, j are both approximately equivalent. The transition probabilities into and out of the new state are obtained from the transition probabilities between the aggregated states and the remaining state space. It can be shown that if the aggregated states consist of states that are sufficiently similar to each other, the compressed Markov model satisfies

$$(A^n)_c \simeq (A_c)^n$$

where the subscript c denotes compression operator and A is the original transition matrix. In a more general sense, the "lumpability" of MCs was discussed in [18, 9, 31], where some necessary and sufficient conditions for an MC to be (approximately) lumpable have been proposed. However, there is no practical algorithm which can test the lumpability conditions efficiently.

Compared to MCs, the model compression for the more general class of hidden Markov models (HMMs) is much less well understood as both state transition and observation properties need to be considered. In a number of studies, HMM compression was based on generalizing MC compression [21, 39, 11]. This is done by clustering the hidden states according to the transition matrix as in MCs (often using the NCDMC model) with an additional constraint that states belonging to the same cluster should have similar observation probability distributions. White [45] proposed a definition of lumpability and approximate lumpability of HMMs, but there is no efficient procedure for testing this lumpability. A compression method specifically designed for HMMs proposed in [20, 19] transforms the HMM to a certain class of stochastic jump linear systems (JLS's) and then uses balanced truncation of JLS's to reduce the size of the HMM. For this method, no additional assumptions about the structure of the hidden chain is required. However, result of the compression is a "black box" and does not provide a mapping which can relate the structure of the compressed model to the structure of the original model. Finesso [13] analyzed the Kullback-Leibler (KL) divergence rate between an arbitrary stationary stochastic process and an HMM, and proposed a method to construct an optimally compressed HMM with respect to the divergence rate based approximation criterion for a given stationary stochastic process. However, this method involves an infinite dimensional optimization problem. Although it can be solved using approximate techniques, the computation complexity increases dramatically with the number of states, therefore effectively limiting it to small-to-moderately sized stochastic models.

The objective of this paper is to propose a new compression method for HMMs based on a probability distance measure. Recently, probability distances, which can measure the difference between probability measures, have received increased interest, because they allow one to decide whether the proposed stochastic model is a satisfactory approximation to the real model [15]. In this paper, we derive an easily

computable criterion for the HMM compression based on the total variation (TV) distance [24] between the observation sequences of HMMs, and develop a smoothing approximation based search strategy to optimize this criterion. In contrast to previous approaches, the present compression method does not assume a specific structure of the original model, such as NCDMC or similar transient behaviors [41, 40]. Moreover the compressed model can be directly related to the structure of the original model via a fuzzy partition [14].

Notation. Let $\mathbb{R}$, $\mathbb{R}^n$ and $\mathbb{R}^{m\times n}$ denote the sets of real numbers, real n-vectors and real $m \times n$ matrices, respectively. The cardinality of a set S is denoted by $|S|$. Given a matrix $G = [g_{ij}] \in \mathbb{R}^{m\times n}$, the 1-norm of G is defined by

$$\|G\|_1 = \max_j \sum_i |g_{ij}| \tag{1.1}$$

And for a vector $g = (g_1, \ldots, g_n)$,

$$\|g\|_1 = \sum_i |g_i| \tag{1.2}$$

$$\|g\|_2 = \sqrt{\sum_i g_i^2} \tag{1.3}$$

The notation $G \succeq 0$ stands for each element of G being nonnegative. $\mathbf{0}$ and $\mathbf{1}$ denote the column vectors of zeros and ones of appropriate size. Given a sequence $\{x_t | t = 0, 1, 2, \ldots\}$, we denote the set $\{x_t | k \le t \le l\}$ by $x_{k:l}$. Uniform distribution on the set S is denoted as $\mathcal{U}_S$. And $1_{\{x=y\}}$ is defined by

$$1_{\{x=y\}} = \begin{cases} 1, & x = y \\ 0, & x \neq y \end{cases} \tag{1.4}$$

2. Background.

2.1. Total variation distance for probability distributions. There are various possibilities to measure distances between probability distributions [15]. Here we introduce the total variation (TV) distance. Suppose μ, ν are two probability distributions on a space Ω, the TV distance between μ and ν is defined by

$$d_{TV}(\mu, \nu) = \sup_{A\subset\Omega} |\mu(A) - \nu(A)| \tag{2.1}$$

For a finite state space $\Omega = \{1, \ldots n\}$, this can be easily computed as:

$$d_{TV}(\mu, \nu) = \frac{1}{2}\sum_{i=1}^{n} |\mu(i) - \nu(i)| = \frac{1}{2}\|\bar{\mu} - \bar{\nu}\|_1 \tag{2.2}$$

where $\bar{\mu} = (\mu(1), \ldots, \mu(n))$, $\bar{\nu} = (\nu(1), \ldots, \nu(n))$. It is obvious that $d_{TV}(\mu, \nu) = 0$ iff $\mu = \nu$.

2.2. Hidden Markov models. HMMs are extensively described in many publications (e.g. [35, 34, 12]). We just state the HMM nomenclature used in this paper. A HMM is a stochastic finite state machine, specified by a 5-tuple $\mathcal{H} = (\mathcal{S}, \mathcal{O}, A, B, \pi)$ where

- $\mathcal{S}$ is the set of hidden states with cardinality N. We denote the individual states as $\mathcal{S} = \{1, \ldots, N\}$, and the state at time t as s_t.
- $\mathcal{O} = \{1, \ldots, M\}$ is the set of observations and the observation at time t is denoted by o_t.
- $A = [a_{ij}] \in \mathbb{R}^{N \times N}$ is the state transition matrix with probabilities

$$a_{ij} = p\left(s_{t+1} = i | s_t = j\right)$$

- $B = [b_{kj}] \in \mathbb{R}^{M \times N}$ is the observation probability matrix

$$b_{kj} = p\left(o_t = k | s_t = j\right)$$

- $\pi = [\pi_i] \in \mathbb{R}^N$ is the probability distribution for the initial state

$$\pi_i = p\left(s_0 = i\right)$$

Remark 2.1. In this paper, we only consider the case that A and B are both constant, *i.e.*, the HMM is time-homogenous.

For a given HMM $\mathcal{H} = (\mathcal{S}, \mathcal{O}, A, B, \pi)$, the probability distribution of all available observations $o_{1:T}$ up to time T is

$$\begin{aligned} p\left(o_{1:T}\right) &= \sum_{s_{0:T}} \left(\prod_{t=1}^{T} p\left(o_t | s_t\right) p\left(s_t | s_{t-1}\right) \right) p\left(s_0\right) \\ &= \sum_{s_{0:T}} \left(\prod_{t=1}^{T} b_{o_t, s_t} a_{s_t, s_{t-1}} \right) \pi_{s_0} \end{aligned} \tag{2.3}$$

In practice, the above $p\left(o_{1:T}\right)$ defined in (2.3) can be calculated through the forward-backward procedure [34] as

$$p\left(o_{1:T}\right) = \mathbf{1}^{\mathrm{T}} w_{o_{1:T}}\left(\mathcal{H}\right) \tag{2.4}$$

where

$$w_{o_{1:t}}\left(\mathcal{H}\right) = P_{o_t}\left(A, B\right) \ldots P_{o_1}\left(A, B\right) \pi \tag{2.5}$$

and $P_k\left(A, B\right)$ are observable operators defined as

$$P_k\left(A, B\right) = \begin{bmatrix} b_{k1} & & \\ & \ddots & \\ & & b_{kN} \end{bmatrix} A, \quad 1 \leq k \leq M \tag{2.6}$$

Moreover, it is easy to verify that $w_{o_{1:t}}\left(\mathcal{H}\right)$ and $P_k\left(A, B\right)$ satisfy the following equations

$$w_{o_{1:t}}\left(\mathcal{H}\right) = \begin{bmatrix} p\left(s_t = 1, o_{1:t}\right) \\ \vdots \\ p\left(s_t = N, o_{1:t}\right) \end{bmatrix} \tag{2.7}$$

$$\sum_{o_{1:t}} \left\| w_{o_{1:t}}(\mathcal{H}) \right\|_1 = 1 \tag{2.8}$$

and

$$\left\| P(A,B) \right\|_1 = 1 \tag{2.9}$$

where

$$P(A,B) = \begin{bmatrix} P_1(A,B) \\ \vdots \\ P_M(A,B) \end{bmatrix} \tag{2.10}$$

3. HMM Compression.

3.1. TV distance for HMMs. Based on the above definitions, the set of all observations up to time t generated by an HMM with observation set $\mathcal{O}$ can be treated as a random variable on the space $\mathcal{O}^t$, and its probability distribution is given by (2.4). Therefore, the similarity between two HMMs $\mathcal{H}^1 = \left(\mathcal{S}^1, \mathcal{O}, A^1, B^1, \pi^1\right)$ and $\mathcal{H}^2 = \left(\mathcal{S}^2, \mathcal{O}, A^2, B^2, \pi^2\right)$ with the same observation set can be measured by calculating the distance between the distributions of $o_{1:t}|\mathcal{H}^1$ and $o_{1:t}|\mathcal{H}^2$ for different t. However, explicit evaluation of the full distributions $o_{1:t}|\mathcal{H}^1$, $o_{1:t}|\mathcal{H}^2$ is very expensive for long t and is not practical as a part of an optimization procedure. Therefore, we instead consider the following function which can be computed quickly and can be proved to provide an upper bound to the TV distance between two HMM outputs:

$$D_{TV}\left(\mathcal{H}^1, \mathcal{H}^2, R\right) = \left\| P\left(A^1, B^1\right) R - \left(I_M \otimes R\right) P\left(A^2, B^2\right) \right\|_1 \tag{3.1}$$

where R is a parameter matrix which satisfies $\mathbf{1}^{\mathrm{T}} R = \mathbf{1}^{\mathrm{T}}$, $I_M \in \mathbb{R}^{M \times M}$ is the identity matrix and $\otimes$ denotes the Kronecker product. In our case, R will turn out to be the compression operator which is explained in detail below. First, let us summarize a number of important facts about $D_{TV}(\cdot)$:

THEOREM 3.1. *Let $\mathcal{H}^1 = \left(\mathcal{S}^1, \mathcal{O}, A^1, B^1, \pi^1\right)$ and $\mathcal{H}^2 = \left(\mathcal{S}^2, \mathcal{O}, A^2, B^2, \pi^2\right)$ be two HMMs with $\left|\mathcal{S}^1\right| = N^1 \leq \left|\mathcal{S}^2\right| = N^2, |\mathcal{O}| = M$. Then $D_{TV}\left(\mathcal{H}^1, \mathcal{H}^2, R\right)$ defined in (3.1) satisfies the following properties:*

1. *For any $t \geq 1$, we have*

$$\begin{aligned} \sum_{o_{1:t}} \left\| w_{o_{1:t}}\left(\mathcal{H}^1\right) - R w_{o_{1:t}}\left(\mathcal{H}^2\right) \right\|_1 \leq & \sum_{o_{1:t-1}} \left\| w_{o_{1:t-1}}\left(\mathcal{H}^1\right) - R w_{o_{1:t-1}}\left(\mathcal{H}^2\right) \right\|_1 \\ & + D_{TV}\left(\mathcal{H}^1, \mathcal{H}^2, R\right) \end{aligned} \tag{3.2}$$

where $w_{o_{1:0}}\left(\mathcal{H}^1\right)$ and $w_{o_{1:0}}\left(\mathcal{H}^2\right)$ are set to π^1 and π^2.

2. *The TV distance between $o_{1:t}|\mathcal{H}^1$ and $o_{1:t}|\mathcal{H}^2$ remains bounded as*

$$\begin{aligned} 2 d_{TV}\left(o_{1:t}|\mathcal{H}^1, o_{1:t}|\mathcal{H}^2\right) & \leq \sum_{o_{1:t}} \left\| w_{o_{1:t}}\left(\mathcal{H}^1\right) - R w_{o_{1:t}}\left(\mathcal{H}^2\right) \right\|_1 \\ & \leq \left\| \pi^1 - R\pi^2 \right\|_1 + t D_{TV}\left(\mathcal{H}^1, \mathcal{H}^2, R\right) \end{aligned} \tag{3.3}$$

3. *There exists a matrix R such that $\mathbf{1}^{\mathrm{T}} R = \mathbf{1}^{\mathrm{T}}$ and*

$$D_{TV}\left(\mathcal{H}^1, \mathcal{H}^2, R\right) = 0 \tag{3.4}$$

if $d_{TV}\left(o_{1:t}|\mathcal{H}^1, o_{1:t}|\mathcal{H}^2\right) = 0$ for any $t \geq 1$.

Proof.

Part (1): According to (2.5), for a given observation sequence $o_{1:t-1}$,

$$\sum_{o_t} \left\| w_{o_{1:t}}\left(\mathcal{H}^1\right) - R w_{o_{1:t}}\left(\mathcal{H}^2\right) \right\|_1 = \left\| P\left(A^1, B^1\right) w_{o_{1:t-1}}\left(\mathcal{H}^1\right) - \left(I_M \otimes R\right) P\left(A^2, B^2\right) w_{o_{1:t-1}}\left(\mathcal{H}^2\right) \right\|_1 \tag{3.5}$$

For simplicity, we denote $P^1 = P\left(A^1, B^1\right)$, $w^1_{o_{1:t}} = w_{o_{1:t}}\left(\mathcal{H}^1\right)$ and $P^2 = P\left(A^2, B^2\right)$, $w^2_{o_{1:t}} = w_{o_{1:t}}\left(\mathcal{H}^2\right)$. Then

$$\begin{aligned}
\sum_{o_t} \left\| w^1_{o_{1:t}} - R w^2_{o_{1:t}} \right\|_1 &= \left\| P^1 w^1_{o_{1:t-1}} - \left(I_M \otimes R\right) P^2 w^2_{o_{1:t-1}} \right\|_1 \\
&= \left\| P^1 w^1_{o_{1:t-1}} - P^1 R w^2_{o_{1:t-1}} + P^1 R w^2_{o_{1:t-1}} - \left(I_M \otimes R\right) P^2 w^2_{o_{1:t-1}} \right\|_1 \\
&\leq \left\| P^1 w^1_{o_{1:t-1}} - P^1 R w^2_{o_{1:t-1}} \right\|_1 \\
&\quad + \left\| P^1 R w^2_{o_{1:t-1}} - \left(I_M \otimes R\right) P^2 w^2_{o_{1:t-1}} \right\|_1 \\
&\leq \left\| P^1 \right\|_1 \left\| w^1_{o_{1:t-1}} - R w^2_{o_{1:t-1}} \right\|_1 \\
&\quad + \left\| P^1 R - \left(I_M \otimes R\right) P^2 \right\|_1 \left\| w^2_{o_{1:t-1}} \right\|_1
\end{aligned} \tag{3.6}$$

By summing both sides of equation (3.6) with respect to $o_{1:t-1}$ and using (2.8), (2.9), we get

$$\begin{aligned}
\sum_{o_{1:t}} \left\| w^1_{o_{1:t}} - R w^2_{o_{1:t}} \right\|_1 &\leq \sum_{o_{1:t-1}} \left\| w^1_{o_{1:t-1}} - R w^2_{o_{1:t-1}} \right\|_1 + \left\| P^1 R - \left(I_M \otimes R\right) P^2 \right\|_1 \\
&= \sum_{o_{1:t-1}} \left\| w^1_{o_{1:t-1}} - R w^2_{o_{1:t-1}} \right\|_1 + D_{TV}\left(\mathcal{H}^1, \mathcal{H}^2, R\right)
\end{aligned} \tag{3.7}$$

Part (2): From (2.4) and TV distance definition (2.2) we have

$$\begin{aligned}
2 d_{TV}\left(o_{1:t}|\mathcal{H}^1, o_{1:t}|\mathcal{H}^2\right) &= \sum_{o_{1:t}} \left| \mathbf{1}^{\mathrm{T}} w_{o_{1:t}}\left(\mathcal{H}^1\right) - \mathbf{1}^{\mathrm{T}} w_{o_{1:t}}\left(\mathcal{H}^2\right) \right| \\
&= \sum_{o_{1:t}} \left| \mathbf{1}^{\mathrm{T}} \left(w_{o_{1:t}}\left(\mathcal{H}^1\right) - R w_{o_{1:t}}\left(\mathcal{H}^2\right) \right) \right| \\
&\leq \sum_{o_{1:t}} \left\| w_{o_{1:t}}\left(\mathcal{H}^1\right) - R w_{o_{1:t}}\left(\mathcal{H}^2\right) \right\|_1
\end{aligned} \tag{3.8}$$

Using the definition of $w_{o_{1:0}}$ and summing relation (3.2) from 1 to t, we obtain

$$\sum_{o_{1:t}} \left\| w_{o_{1:t}}\left(\mathcal{H}^1\right) - R w_{o_{1:t}}\left(\mathcal{H}^2\right) \right\|_1 \leq \left\| \pi^1 - R\pi^2 \right\|_1 + t D_{TV}\left(\mathcal{H}^1, \mathcal{H}^2, R\right) \tag{3.9}$$

Part (3) is an immediate consequence of equivalence theorems for observable operator models (OOMs) (see Propositions 14, 15 and 16 in [17]). □

This theorem shows that the operator $D_{TV}(\cdot)$ can be used to evaluate quantitatively the differences between dynamics of HMMs.

3.2. Optimal compression model. Based on the function D_{TV} we can now develop a framework for compressing HMMs.

First of all, it is important to define what is meant by "perfect" compression. Let $\mathcal{H}^0 = \left(\mathcal{S}^0, \mathcal{O}, A^0, B^0, \pi^0\right)$ be the original HMM and $\mathcal{H}^c = (\mathcal{S}^c, \mathcal{O}, A^c, B^c, \pi^c)$ be the compressed model obtained by some compression algorithm where $|\mathcal{S}^c| = N^c < \left|\mathcal{S}^0\right| = N^0$ and $|\mathcal{O}| = M$. Since both $\mathcal{H}^0$ and $\mathcal{H}^c$ generate observation sequences from observation set $\mathcal{O}$, the output of $\mathcal{H}^c$ should have (nearly) the same distribution as that of $\mathcal{H}^0$. In other words, perfect compression has occurred if

$$d_{TV}\left(o_{1:t}|\mathcal{H}^c, o_{1:t}|\mathcal{H}^0\right) = 0, \quad \forall o_{1:t} \in \mathcal{O}^t, t \geq 1 \tag{3.10}$$

However, (3.10) cannot be satisfied in most cases for $N^c < N^0$. As an alternative, we compress the original model through minimizing the upper bound of $d_{TV}\left(o_{1:t}|\mathcal{H}^c, o_{1:t}|\mathcal{H}^0\right)$ presented in Theorem 3.1:

$$\begin{aligned} d_{TV}\left(o_{1:t}|\mathcal{H}^c, o_{1:t}|\mathcal{H}^0\right) &\leq \frac{1}{2}\left\|\pi^c - R^c\pi^0\right\|_1 + \frac{t}{2}D_{TV}\left(\mathcal{H}^c, \mathcal{H}^0, R^c\right) \\ &= \frac{t}{2}\left(\frac{1}{t}\left\|\pi^c - R^c\pi^0\right\|_1 + D_{TV}\left(\mathcal{H}^c, \mathcal{H}^0, R^c\right)\right) \end{aligned} \tag{3.11}$$

Note that the upper bound is a positive linear combination of $D_{TV}\left(\mathcal{H}^c, \mathcal{H}^0, R^c\right)$ and the "initial TV distance" $\left\|\pi^c - R^c\pi^0\right\|_1/2$, and the value of $D_{TV}\left(\mathcal{H}^c, \mathcal{H}^0, R\right)$ is independent of π^c. Considering that for a given $(\mathcal{H}^c, R^c)$,

$$\frac{1}{t}\left\|\pi^c - R^c\pi^0\right\|_1 + D_{TV}\left(\mathcal{H}^c, \mathcal{H}^0, R^c\right) \simeq D_{TV}\left(\mathcal{H}^c, \mathcal{H}^0, R^c\right)$$

when t is sufficiently large, here a greedy strategy is adopted to optimize the compression model. First, A^c, B^c, R^c are obtained by minimizing $D_{TV}\left(\mathcal{H}^c, \mathcal{H}^0, R^c\right)$, *i.e.*,

$$\begin{aligned} &(A^c, B^c, R^c) = \arg\min_{A,B,R}\left\|P\left(A, B\right)R - \left(I_M \otimes R\right)P\left(A^0, B^0\right)\right\|_1 \\ &\text{subject to:} \\ &\quad \mathbf{1}^{\mathrm{T}}A = \mathbf{1}^{\mathrm{T}}, \mathbf{1}^{\mathrm{T}}B = \mathbf{1}^{\mathrm{T}}, \mathbf{1}^{\mathrm{T}}R = \mathbf{1}^{\mathrm{T}} \\ &\quad A, B \succeq 0 \end{aligned} \tag{3.12}$$

then the initial distribution π^c is obtained by

$$\begin{aligned} &\pi^c = \arg\min_{\pi}\left\|\pi - R^c\pi^0\right\|_1 \\ &\text{subject to:} \\ &\quad \mathbf{1}^{\mathrm{T}}\pi = 1, \pi \succeq 0 \end{aligned} \tag{3.13}$$

In this paper, we denote the solutions of (3.12), (3.13) by

$$(\mathcal{H}^c, R^c) = OCH\left(\mathcal{H}^0, N^c\right).$$

Remark 3.2. Obviously the greedy strategy is a "suboptimal method" since the optimization problems (3.12) and (3.13) are coupled via R^c. But it is easy to show that the solution $(\mathcal{H}^c, R^c) = OCH\left(\mathcal{H}^0, N^c\right)$ also satisfies

$$(\mathcal{H}^c, R^c) = \arg\min_{\mathcal{H},R}\frac{1}{t}\left\|\pi - R\pi^0\right\|_1 + D_{TV}\left(\mathcal{H}, \mathcal{H}^0, R\right)$$

with $\pi^c = R^c\pi^0$ for any $t > 0$ if $R^c\pi^0 \succeq 0$. More specifically, if $R^c \succeq 0$, which means the dynamics of $\mathcal{H}^0$ can be well compressed through a fuzzy partition on $\mathcal{S}^0$ (see Subsection 3.3), $R^c\pi^0 \succeq 0$ must hold.

Remark 3.3. An important issue for solving the compression problem is determining the value of N^c. One possible approach is to run the compression algorithm many times with different N^c, and choose a best value according to a user-defined criterion $C\left(N^c, D_{TV}\left(\mathcal{H}^c, \mathcal{H}^0, R^c\right)\right)$ (e.g. $D_{TV}\left(\mathcal{H}^c, \mathcal{H}^0, R^c\right) + w_r N^c$ with $w_r > 0$). Certainly, $N^{c1} > N^{c2}$ implies that $D_{TV}\left(\mathcal{H}^{c1}, \mathcal{H}^0, R^{c1}\right) < D_{TV}\left(\mathcal{H}^{c2}, \mathcal{H}^0, R^{c2}\right)$, but depending on the application there may be a tolerable upper bound to N^c. Here, we do not address the problem of choosing the number of compressed hidden states in this paper, and assume that N^c is given.

3.3. Optimal compression based fuzzy partition. In contrast to typical MC compression methods, the proposed optimal compression model does not involve any explicit state aggregation. However we can infer a fuzzy partition over the set of original hidden states from the optimal compression model.

Suppose the model compression has been performed well, then

$$D_{TV}\left(\mathcal{H}^c, \mathcal{H}^0, R^c\right) \simeq 0 \tag{3.14}$$

and

$$\left\|\pi - R^c\pi^0\right\|_1 \simeq 0 \tag{3.15}$$

From (3.3) we have

$$w_{o_{1:t}}\left(\mathcal{H}^c\right) \simeq R^c w_{o_{1:t}}\left(\mathcal{H}^0\right) \tag{3.16}$$

That is to say, the joint probability distribution of the hidden states and observations in the compressed model can be approximated by

$$p\left(s_t^c = i, o_{1:t}|\mathcal{H}^c\right) \simeq \sum_{j=1}^{N^0} R_{ij}^c p\left(s_t^0 = j, o_{1:t}|\mathcal{H}^0\right) \tag{3.17}$$

where R_{ij}^c denotes the element in the i-th row and j-th column of R^c, s_t^0, s_t^c are original and compressed hidden states at time t.

On the other hand, if we define a fuzzy partition on $\mathcal{S}^0$ by N^c fuzzy sets $\tilde{A}_1$, ..., $\tilde{A}_{N^c}$ with membership functions

$$m_{\tilde{A}_i}(j) = \mu_{ij} \tag{3.18}$$

and a probability distribution on $\mathcal{S}^0$, the probability distribution of fuzzy sets and observations can be expressed as [46, 14]

$$p\left(s_t^0 \in \tilde{A}_i, o_{1:t}|\mathcal{H}^0\right) = \sum_{j=1}^{N^0} \mu_{ij} p\left(s_t^0 = j, o_{1:t}|\mathcal{H}^0\right) \tag{3.19}$$

Comparing (3.17) and (3.19), we can conclude that R^c represents a fuzzy partition on $\mathcal{S}^0$ with membership function

$$m_{\tilde{A}_i}(j) = R_{ij}^c \tag{3.20}$$

which plays a similar to the "almost characteristic functions" in robust PCCA [10], and the dynamics of hidden states in $\mathcal{H}^c$ is approximately equivalent to that of fuzzy hidden states in $\mathcal{H}^0$, if $R^c \succeq 0$.

3.4. Compression algorithm. In the optimal compression model, it is easy to convert (3.13) into a linear programming model problem that can be solved using standard software packages. So we only discuss how to solve (3.12).

3.4.1. Smooth approximation. The main difficulty in solving the optimal compression model (3.12) is that the objective function involves computation of matrix 1-norm which is not differentiable. In this subsection, we will propose a smoothing strategy to overcome this problem, that takes advantage of the "soft maximum" operator [32]

$$f_\gamma^s(x) = \gamma \log \sum_{j=1}^{k} \exp\left(\frac{x_j}{\gamma}\right) \tag{3.21}$$

where $x = (x_1, \ldots, x_k)$ and $\gamma > 0$ is the smoothing parameter. The function $f_\gamma^s(\cdot)$ is twice continuous differentiable and provides a good approximation of $\max_k x_k$ with a small γ in the sense that [32]

$$\max_k x_k \le f_\gamma^s(x) \le \max_k x_k + \gamma \log k \tag{3.22}$$

Replacing the maximum operation in (1.1) by $f_\gamma^s(\cdot)$ (note that the absolute value function $|x|$ can also be written as $\max\{x, -x\}$), we can approximate the matrix 1-norm as

$$\|G\|_1 \simeq F_\gamma^s(G) = \gamma \log \sum_{j=1}^{n} \exp\left(\sum_{i=1}^{m} \log\left(\exp\left(\frac{g_{ij}}{\gamma}\right) + \exp\left(-\frac{g_{ij}}{\gamma}\right)\right)\right) \tag{3.23}$$

for $G = [g_{ij}] \in \mathbb{R}^{m\times n}$. It is easy to prove that

$$\|G\|_1 \le F_\gamma^s(G) \le \|G\|_1 + \gamma(\log n + m \log 2) \tag{3.24}$$

from (3.22).

Therefore, (3.12) can be approximately solved by minimizing the smooth objective function

$$D_\gamma^s(A, B, R) = F_\gamma^s\left(P(A, B)R - (I_M \otimes R)P\left(A^0, B^0\right)\right) \tag{3.25}$$

with a decreasing sequence of γ. The sequential optimization algorithm is then described as follows:

Let the numbers $\beta \in (0, 1)$, $\gamma_1 > 0$ and a small enough number $\epsilon_s > 0$ be given.

Step 1 Select an initial feasible solution $\left(A^{(1)}, B^{(1)}, R^{(1)}\right)$. Set $k = 1$.

Step 2 Perform a local search algorithm to compute

$$\begin{aligned} &\left(A^{(k+1)}, B^{(k+1)}, R^{(k+1)}\right) = \arg\min_{A,B,R} D_{\gamma_k}^s(A, B, R) \\ &\text{subject to:} \\ &\quad \mathbf{1}^{\mathrm{T}} A = \mathbf{1}^{\mathrm{T}}, \mathbf{1}^{\mathrm{T}} B = \mathbf{1}^{\mathrm{T}}, \mathbf{1}^{\mathrm{T}} R = \mathbf{1}^{\mathrm{T}} \\ &\quad A, B \succeq 0 \end{aligned} \tag{3.26}$$

starting at $\left(A^{(k)}, B^{(k)}, R^{(k)}\right)$.

Step 3 Terminate if

$$\gamma_k\left(\log N^0 + MN^c \log 2\right) \le \epsilon_s \tag{3.27}$$

Step 4 Let $\gamma_{k+1} = \beta\gamma_k$ and go to Step 2 with k replaced by $k+1$.

Remark 3.4. In Step 3 the condition (3.27) guarantees that

$$\left|D^s_{\gamma_k}(A,B,R) - D_{TV}\left(\mathcal{H},\mathcal{H}^0,R\right)\right| \leq \epsilon_s$$

Remark 3.5. In Step 2, the subproblem (3.26) is a smooth optimization problem with linear constraints, and a variety of methods can be used to search the solution. Here we select the projected gradient algorithm (PGA) proposed in [6] (see Appendix A for details).

3.4.2. Heuristic search. Note that (3.12) is an optimization problem with $N^c\left(N^c+M+N^0\right)$ variables, and A, B and R are strongly coupled. The algorithm in Subsection 3.4.1 might be inefficient and get stuck in local minima for large M or N^0. To avoid this problem, we propose a heuristic search strategy for finding a good initial solution, which reduces the solution space and improves the global search ability by utilizing the relationship between optimal compression models and fuzzy partitions. It consists of the following steps:

Step 1 Suppose that R can be described by N^c parametric fuzzy sets $\tilde{A}_1,\ldots,\tilde{A}_{N^c}$ as

$$R_{ij}(\theta) = m_{\tilde{A}_i}(j|\theta) \tag{3.28}$$

where θ is a parameter vector.

Step 2 Suppose the compressed model is approximately equivalent to the fuzzy partition model, then $A(\theta), B(\theta)$ can be calculated as

$$\begin{aligned} a_{ij}(\theta) &\propto p\left(s^0_{t+1}\in\tilde{A}_i, s^0_t\in\tilde{A}_j|\mathcal{H}^0, s^0_t\sim\pi^s,\theta\right) \\ &\propto \sum_{k,l}\pi^s_l a^0_{kl}R_{ik}(\theta)R_{jl}(\theta) \end{aligned} \tag{3.29}$$

and

$$\begin{aligned} b_{kj}(\theta) &\propto p\left(o_t=k, s^0_t\in\tilde{A}_j|\mathcal{H}^0,, s^0_t\sim\pi^s,\theta\right) \\ &\propto \sum_{l}\pi^s_l b^0_{kl}R_{jl}(\theta) \end{aligned} \tag{3.30}$$

where $\pi^s=[\pi^s_i]$ denotes the stationary distribution of A^0, $A^0=\left[a^0_{ij}\right]$, $B^0=\left[b^0_{kj}\right]$.

Step 3 Solve

$$\theta^* = \arg\min_\theta D^s_{\gamma_1}(A(\theta),B(\theta),R(\theta)) \tag{3.31}$$

Step 4 Perform the algorithm in Subsection 3.4.1 to solve (3.12) with initial feasible solution $\left(A^{(1)},B^{(1)},R^{(1)}\right)=(A(\theta^*),B(\theta^*),R(\theta^*))$.

Obviously, the size of problem (3.31) is much smaller than (3.12) since $\dim(\theta)=O(N^c)$ in most cases. Therefore we can often find a suboptimal but satisfactory compression model quickly by solving (3.31).

Remark 3.6. $R(\theta)$ in Step 1 can be defined according to the actual situation of $\mathcal{H}^0$. For example, assume that each element of $\mathcal{S}^0$ can be associated with a coordinate in $\mathbb{R}^{n_{HS}}$, we can define $R(\theta)$ by the following fuzzy partition model [4]:

$$R_{ij}(\theta) = m_{\tilde{A}_i}(j|\theta) = \frac{\left\|x^j - z^i\right\|_2^{-2}}{\sum_{k=1}^{N^c} \left\|x^j - z^k\right\|_2^{-2}} \tag{3.32}$$

where x^j denotes the coordinate of the j-th element of $\mathcal{S}^0$, z^i is the center of i-th fuzzy set of the fuzzy partition, and $\theta = \left(z^1, \ldots, z^{N^c}\right)$.

Remark 3.7. π^s in Step 3 is just a weight vector for the fuzzy combination, which can be replaced with some other vector as needed. For example, we can set [29]

$$\pi_i^s = \lim_{T\to\infty} \frac{1}{T} \mathbb{E}\left[\sum_{t=0}^{T} 1_{\{s_t=i\}}\right] \tag{3.33}$$

if A^0 has multiple stationary distributions.

Remark 3.8. In this paper, (3.31) is solved by the estimation of distribution algorithm (EDA), which is a global optimization algorithm and whose effectiveness have been evaluated by many works [23]. See Appendix B for details.

4. Numerical Experiments. In this section, the proposed compression proposed in this paper will be applied to three examples of HMM models. The algorithm parameters are chosen as

$$\begin{array}{l}\beta = 0.5, \gamma_1 = 1, \epsilon_s = 10^{-4}, \mu_{PG} = 0.1, \gamma_{PG} = 1, \epsilon_{PG} = 10^{-6},\\ P_p = 500, P_{se} = 250, K_{\max} = 4000\end{array}$$

4.1. Quasi-lumpable model. Consider an HMM $\mathcal{H}^0$ with $N^0 = 7$ and $M = 2$ characterized by

$$A^0 = \begin{bmatrix} * & 0 & 0.4+\epsilon & 0.2 & 0 & 0 & 0 \\ 0 & * & 0 & 0.2 & 0.4 & 0 & 0 \\ 0.4+\epsilon & 0 & * & 0 & 0 & 0.4 & 0 \\ 0.4 & 0.4 & 0 & * & 0 & 0.2 & 0.4 \\ 0 & 0.4 & 0 & 0 & * & 0 & 0.2 \\ 0 & 0 & 0.4+\epsilon & 0.0 & 0.4 & * & 0 \\ 0 & 0 & 0 & 0.4 & 0 & 0 & * \end{bmatrix} \tag{4.1}$$

$$B^0 = \begin{bmatrix} 0.8 & 0.8+\epsilon & 0.5 & 0.5-\epsilon & 0.5+\epsilon & 0.3 & 0.3+\epsilon \\ * & * & * & * & * & * & * \end{bmatrix} \tag{4.2}$$

and π^0 equals to the stationary distribution of A^0, where elements $*$ make each column of matrices sum up to 1. In this example, the transition matrix A^0 is ϵ-quasi-lumpable with respect to the partition $\mathcal{P} = \{S_1, S_2, S_3\} = \{\{1,2\}, \{3,4,5\}, \{6,7\}\}$ (see Section 6.1 in [31] for details), and it is easy to verify that $\mathcal{H}^0$ is a lumpable HMM if $\epsilon = 0$ according to the lumpability condition proposed in [45].

Here we let $\epsilon = 0.01$, $N^c = 3$, and apply the proposed compression method with the parametric fuzzy partition model in heuristic search procedure defined by

$$R_{ij}(\theta) = m_{\tilde{A}_i}(j|\theta) = \frac{\left|\tilde{R}_{i,j}\right|}{\sum_{k=1}^{3} \left|\tilde{R}_{k,j}\right|} \tag{4.3}$$

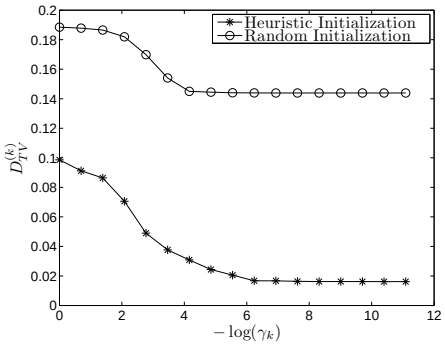


FIG. 4.1. *Optimization results of the smooth approximate problem (3.26) for different* γ_k *where* $D_{TV}^{(k)} = D_{TV}\left(\mathcal{H}^{(k)}, \mathcal{H}^0, R^{(k)}\right)$ *and* $\left(\mathcal{H}^{(k)}, R^{(k)}\right)$ *denotes the solution of the approximate model with smoothing parameter* γ_k.

and $\theta = \left(\tilde{R}_{1,1}, \tilde{R}_{1,2}, \ldots, \tilde{R}_{N^c,N^0}\right)$. For the sake of comparison, we also perform the smooth approximation based optimization algorithm starting with a random initial solution. Fig. 4.1 plots the optimization results under different initialization schemes. As observed from the figure, the randomly generated initial solution leads to the algorithm getting stuck in a local optimum. It shows that the sequential optimization algorithm proposed in Subsection 3.4.1 is sensitive to initial conditions as a local search algorithm, and some global method (e.g. the heuristic search algorithm presented in this paper) is needed to provide a satisfactory initial solution.

The obtained optimal compression results are given by

$$R^c = \begin{bmatrix} 0.9667 & 0.9540 & 0.0045 & 0.0172 & 0.0123 & 0.0138 & 0.0010 \\ 0.0223 & 0.0387 & 0.9800 & 0.9602 & 0.9572 & 0.0193 & 0.0216 \\ 0.0110 & 0.0073 & 0.0155 & 0.0226 & 0.0305 & 0.9670 & 0.9774 \end{bmatrix} \tag{4.4}$$

$$A^c = \begin{bmatrix} 0.1951 & 0.4071 & 0.0000 \\ 0.8049 & 0.1912 & 0.5962 \\ 0.0000 & 0.4017 & 0.4038 \end{bmatrix} \tag{4.5}$$

$$B^c = \begin{bmatrix} 0.7948 & 0.4983 & 0.3091 \\ 0.2052 & 0.5017 & 0.6909 \end{bmatrix} \tag{4.6}$$

Obviously, the fuzzy partition $\left\{\tilde{A}_1, \tilde{A}_2, \tilde{A}_3\right\}$ defined by R^c is consistent with the partition $\mathcal{P}$. We generate an observation sequence $o_{1:T}$ randomly by $\mathcal{H}^0$ with $T = 1000$, and apply both $\mathcal{H}^0$ and $\mathcal{H}^c$ to calculating the likelihood $p\left(o_{1:t}\right)$ and the conditional distribution of hidden state $p\left(s_t|o_{1:t}\right)$ for different t. Fig. 4.2 shows the relative error between $\log p\left(o_{1:t}|\mathcal{H}^c\right)$ and $\log p\left(o_{1:t}|\mathcal{H}^0\right)$, where

$$E_{ll} = \frac{\left|\log p\left(o_{1:t}|\mathcal{H}^c\right) - \log p\left(o_{1:t}|\mathcal{H}^0\right)\right|}{\left|\log p\left(o_{1:t}|\mathcal{H}^0\right)\right|} \tag{4.7}$$

We also compare the distribution $p\left(s_t^c|o_{1:t}, \mathcal{H}^c\right)$ with $p\left(s_t^0 \in \tilde{A}_i|o_{1:t}, \mathcal{H}^0\right)$ by the dis-

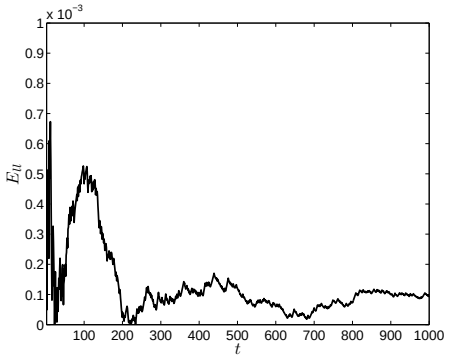


FIG. 4.2. *Relative error of* $\log p\left(o_{1:t}\right)$ *estimated by* $\mathcal{H}^c$

tance

$$E_w = \frac{1}{2}\sum_{i=1}^{N^c}\left|p\left(s_t^c = i|o_{1:t},\mathcal{H}^c\right) - \sum_{j=1}^{N^0} R_{ij}^c p\left(s_t^0 = j|o_{1:t},\mathcal{H}^0\right)\right| \tag{4.8}$$

The mean value of E_w is 4.3044×10^{-3} and the variance is 5.4864×10^{-6} in this simulation. It can be seen that $p\left(o_{1:t}|\mathcal{H}^c\right)$ provides a good estimate of $p\left(o_{1:t}|\mathcal{H}^0\right)$, and the conditional distribution of hidden states s_t^c for given $o_{1:t}$ remains approximately equal to the conditional distribution of fuzzy partitions of the original hidden states.

4.2. Metastable model. In this example a metastable HMM $\mathcal{H}^0$ with 289 states is investigated, where

$$\mathcal{S}^0 = \left\{(x,y)\,|x = -2 + 0.25i, y = -1.5 + 0.25j, i,j = 0,\ldots,16\right\} \tag{4.9}$$

is a grid space,

$$p\left(x_{t+1},y_{t+1}|x_t,y_t,\mathcal{H}^0\right) = \begin{cases} \frac{\min\{1,\exp(V(x_t,y_t)-V(x_{t+1},y_{t+1}))\}}{|N_b(x_t,y_t)|-1}, & (x_{t+1},y_{t+1}) \in N_b\left(x_t,y_t\right)\setminus\left\{(x_t,y_t)\right\} \\ 0, & (x_{t+1},y_{t+1}) \notin N_b\left(x_t,y_t\right) \end{cases} \tag{4.10}$$

defines the transition matrix based on the potential function

$$\begin{aligned} V\left(x,y\right) = {} & 3\exp\left(-\left(x-1\right)^2 - \left(y-1/3\right)^2\right) - 3\exp\left(-\left(x-1\right)^2 - \left(y-5/3\right)^2\right) \\ & -5\exp\left(-\left(x-1\right)^2 - y^2\right) - 3\exp\left(-\left(x+6/5\right)^2 - y^2\right) \end{aligned} \tag{4.11}$$

which is shown in Fig. 4.3,

$$N_b\left(x,y\right) = \left\{\left(x',y'\right)|\,|x'-x| + |y'-y| \le 0.25, \left(x',y'\right) \in \mathcal{S}^0\right\} \tag{4.12}$$

denotes the neighbors of state (x,y), and the output is defined by

$$p\left(o_t|x_t,y_t\right) = \begin{cases} \left(2-x_t\right)/4, & o_t = 1 \\ \left(2+x_t\right)/4, & o_t = 2 \end{cases} \tag{4.13}$$

The stochastic process $(x_{1:t},y_{1:t})$ has three metastable states centered at $(-1.75,0)$, $(1,1.75)$ and $(1,-0.25)$.

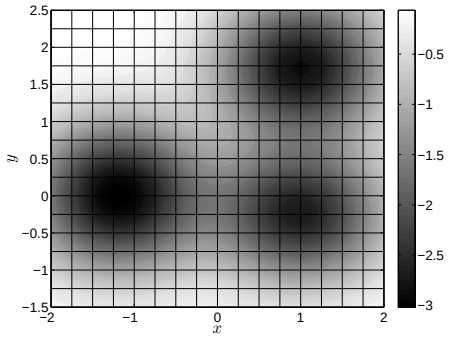
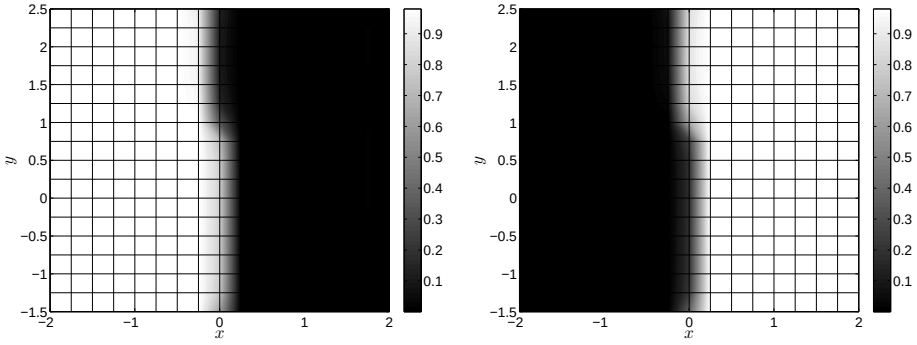


FIG. 4.3. *Potential function* $V(x, y)$



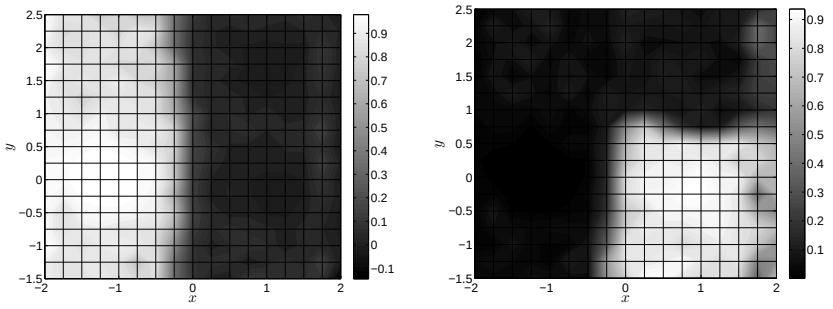
(a) First row (b) Second row

FIG. 4.4. *Values of elements of* R^c *(*$N^c = 2$*)*

We apply the proposed compression method to $\mathcal{H}^0$ with $\theta = \left(x^1, y^1, \ldots, x^{N^c}, y^{N^c}\right)$ and parametric fuzzy sets $\left\{\tilde{A}_1, \ldots, \tilde{A}_{N^c}\right\}$ defined as in Remark 3.6 with coordinates $\{(x, y)\}$, and let $\mathcal{H}_2 = OCH\left(\mathcal{H}^0, 2\right)$, $\mathcal{H}_3 = OCH\left(\mathcal{H}^0, 3\right)$. Figs. 4.4 and 4.5 show R^c with $N^c = 2$ and 3. It can be observed that our method identify the three metastable states successfully when $N^c = 3$. For $N^c = 2$, the metastable states at right side are lumped since they have similar observation probabilities. Figs. 4.6 and 4.7 display the errors of $\mathcal{H}_2$ and $\mathcal{H}_3$ for a $o_{1:T}$ generated by $\mathcal{H}^0$. It is not surprised that the E_{ll} of $\mathcal{H}_3$ is smaller than $\mathcal{H}_2$ because it can capture the dynamics of $\mathcal{H}^0$ more accurately. However, E_w of $\mathcal{H}_3$ is larger $\mathcal{H}_2$, because it is difficult for $\mathcal{H}_3$ to distinguish the right two metastable states in the case that only the observations $o_{1:t}$ are known.

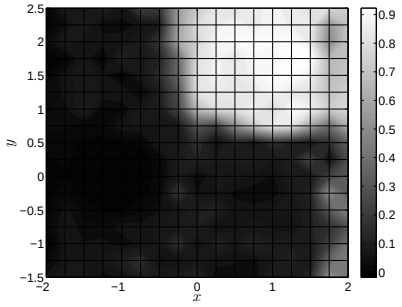
4.3. Traffic model. In this example, we investigate a grid based cellular automaton traffic model [5] of an agent moving in a building as shown in Fig. 4.8, which is similar to the traffic model used in [26]. At each time t, the position of the agent is denoted by (x_t, y_t), *i.e.*, the agent is located in a grid with row x_t and column y_t. If the agent is on a position (x_t, y_t) with a sensor, the agent will be detected with probability 0.9 and report the corridor number. However, the receiver does not know which exact sensor sends the information. The agent movement is Markovian and the detailed dynamics are given in Appendix C.

The objective of this experiment is to estimate which corridor the agent is in at any



(a) First row

(b) Second row



(c) Third row

FIG. 4.5. *Values of elements of* R^c *(*$N^c = 3$*)*

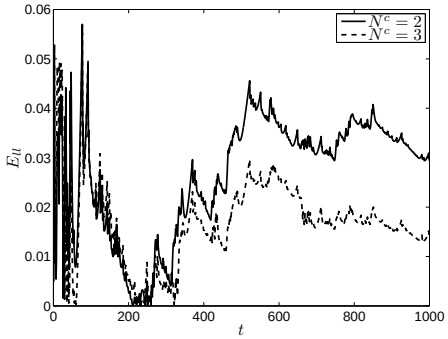


FIG. 4.6. *Relative errors of* $\log p\,(o_{1:t})$ *estimated by* $\mathcal{H}^c$

time based on the past sensor information. According to [25] and [26], the estimation can be performed based on a model $\mathcal{H} = (\mathcal{S}, \mathcal{O}, A, B, \pi)$ with $\mathcal{O} = \{o_1^s, o_2^s, o_3^s, o_1^c, o_2^c, o_3^c\}$ and

$$o_t = \begin{cases} o_i^s, & \text{agent is detected by a sensor in Corridor } i \\ o_i^c, & \text{agent is in Corridor } i \text{ and not detected} \end{cases} \tag{4.14}$$

Here $\mathcal{S}$ only contains the states in the building, thus the exit state is excluded from $\mathcal{S}$, and $\mathcal{H}$ is similar to an HMM except that $\mathbf{1}^{\mathrm{T}} A \neq \mathbf{1}^{\mathrm{T}}$ and $\mathbf{1}^{\mathrm{T}} - \mathbf{1}^{\mathrm{T}} A \succeq 0$, which means that the sequences of s_t and o_t are finite and stop when the agent would hit the exit state (*i.e.* with probability $1 - \sum_i a_{ij}$ at time $t+1$ if s_t is the j-th state of $\mathcal{S}$).

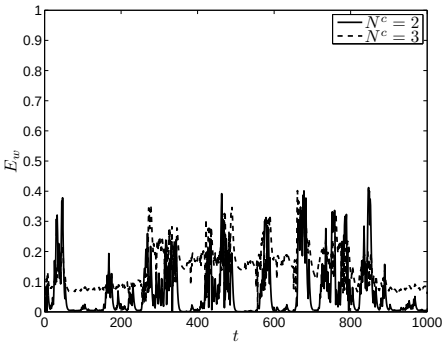


FIG. 4.7. *Distance between* $p\left(s_t^c|o_{1:t},\mathcal{H}^c\right)$ *and* $p\left(s_t^0 \in \tilde{A}_i|o_{1:t},\mathcal{H}^0\right)$

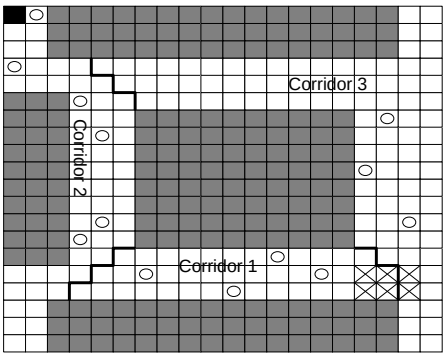


FIG. 4.8. *Grid layout of traffic model where gray squares are obstacles, the black square is the exit,* $\circ$ *are the sensor positions,* $\times$ *indicates the start area, the heavy black lines mark the divisions between the corridors. The agent moves from a random starting square with a random walk until it finds the exit.*

Remark 4.1. Actually, the observations o_1^c, o_2^c, o_3^c can not be distinguished by the sensor information. The sensor observation value is $o_t' \in \{o_1^s, o_2^s, o_3^s, o^c\}$, where $o_t' = o^c$ if no sensor detects the agent.

Let c_t be the number of the corridor where the agent is located at time t. By using $\mathcal{H}$ to approximate the distribution of $o_{1:t}$, we can calculate the Bayesian estimate $p\left(c_t|o_{1:t}'\right)$ online as follows

- If $o_t' = o^c$,

$$\begin{aligned} p\left(c_t|o_{1:t}'\right) &= \sum_{s_t\in\mathcal{S}} p\left(s_t|o_{1:t}'\right) p\left(c_t|s_t, o_{1:t}'\right) \\ &= \sum_{s_t\in\mathcal{S}} p\left(s_t|o_{1:t}'\right) p\left(c_t|s_t, o_t'\right) \\ &= \sum_{s_t\in\mathcal{S}} p\left(s_t|o_{1:t}'\right) \frac{p\left(o_{c_t}^c|s_t\right)}{p\left(o^c|s_t\right)} \end{aligned} \tag{4.15}$$

where $p\left(s_t|o_{1:t}'\right)$ can be calculated recursively by

$$p\left(s_t|o_{1:t}'\right) \propto \sum_{s_{t-1}\in\mathcal{S}} p\left(o_t'|s_t\right) p\left(s_t|s_{t-1}\right) p\left(s_{t-1}|o_{1:t-1}'\right) \tag{4.16}$$

and $p\left(o^c|s_t\right)=\sum_{i=1}^{3}p\left(o_i^c|s_t\right)$.

- If $o'_t=o_i^s$,

$$p\left(c_t|o'_{1:t}\right)=\begin{cases}1, & \text{the agent is in Corridor } c_t\\ 0, & \text{otherwise}\end{cases} \tag{4.17}$$

And the estimate of c_t can be set as $\hat{c}_t=\arg\max_i p\left(c_t=i|o'_{1:t}\right)$.

Clearly, the most accurate model is the "full order" model $\mathcal{H}^0=\left(\mathcal{S}^0,\mathcal{O},A^0,B^0,\pi^0\right)$ where $\mathcal{S}^0$ is the reachable grid set and the elements of A^0,B^0,π^0 are identical to the true probability values of the original traffic model. However, it was found [25, 26] that "reduced order" models $\mathcal{H}^c=(\mathcal{S}^c,\mathcal{O},A^c,B^c,\pi^c)$ obtained by compressing $\mathcal{H}^0$ can also be used in order to save storage space and computation time. We now discuss how to obtain a compressed $\mathcal{H}^c$ with the method proposed in this paper. Considering the sequences of states and observations are both finite for $\mathcal{H}$, we firstly introduce a "null state" s_n and "null observation" o_n. and let

$$\begin{cases}\left(\bar{s}_t,\bar{o}_t\right)=\left(s_t,o_t\right), & t<L\\ \left(\bar{s}_t,\bar{o}_t\right)=\left(s_n,o_n\right), & t\geq L\end{cases} \tag{4.18}$$

where L is the time at which state and observation sequences stop. Then $\mathcal{H}$ can be translated into an equivalent augmented HMM $\bar{\mathcal{H}}=\left(\bar{\mathcal{S}},\bar{\mathcal{O}},\bar{A},\bar{B},\bar{\pi}\right)$ with $\bar{\mathcal{S}}=\mathcal{S}\cup\{s_n\}$, $\bar{\mathcal{O}}=\mathcal{O}\cup\{o_n\}$,

$$\bar{A}=\begin{bmatrix}A & \mathbf{0}\\ * & 1\end{bmatrix},\quad \bar{B}=\begin{bmatrix}B & \mathbf{0}\\ \mathbf{0}^{\mathrm{T}} & 1\end{bmatrix} \tag{4.19}$$

and $\bar{\pi}^{\mathrm{T}}=\left(\pi^{\mathrm{T}},0\right)^{\mathrm{T}}$, where elements $*$ make each column of matrices sum up to 1. It is easy to show that

$$p\left(\bar{o}_{1:t}|\bar{\mathcal{H}}\right)=p\left(o_{1:\min\{L-1,t\}}|\mathcal{H}\right) \tag{4.20}$$

and for two models $\mathcal{H}^1$ and $\mathcal{H}^2$,

$$d_{TV}\left(\bar{o}_{1:t}|\bar{\mathcal{H}}^1,\bar{o}_{1:t}|\bar{\mathcal{H}}^2\right)=d_{TV}\left(o_{1:\min\{L-1,t\}}|\mathcal{H}^1,o_{1:\min\{L-1,t\}}|\mathcal{H}^2\right) \tag{4.21}$$

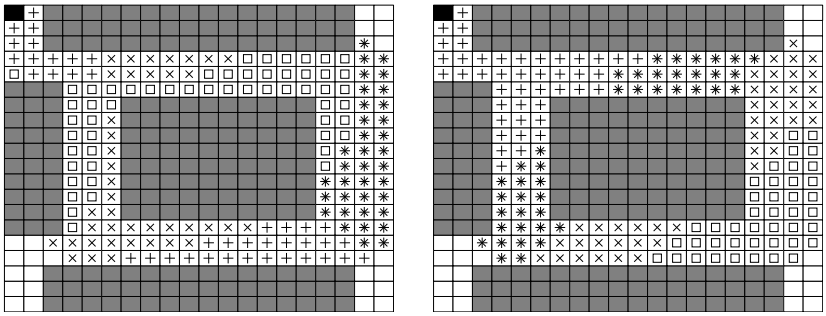
where L is also a random number and $\left\{o_{1:\min\{L-1,t\}}\right\}$ means all the possible observation sequences up to time t (including the sequences that stop by time t). Hence our proposed method can be applied to compressing $\bar{\mathcal{H}}^0=\left(\bar{\mathcal{S}}^0,\bar{\mathcal{O}}^0,\bar{A}^0,\bar{B}^0,\bar{\pi}^0\right)$ and optimizing $\bar{\mathcal{H}}^c=\left(\bar{\mathcal{S}}^c,\bar{\mathcal{O}}^c,\bar{A}^c,\bar{B}^c,\bar{\pi}^c\right)$ with the extra linear constraints shown in (4.19).

Remark 4.2. Because the stationary distribution of the transition matrix $\bar{A}^0$ is $\pi^s=(0,\ldots,0,1)^{\mathrm{T}}$, we replace π^s with the pseudo-stationary distribution $\hat{\pi}^s=[\hat{\pi}_i^s]$ in Step 2 of the heuristic search, where

$$\hat{\pi}_i^s\propto\mathbb{E}\left[\text{Number of times }(x_t,y_t)=i\text{-th grid position until time }L\right] \tag{4.22}$$

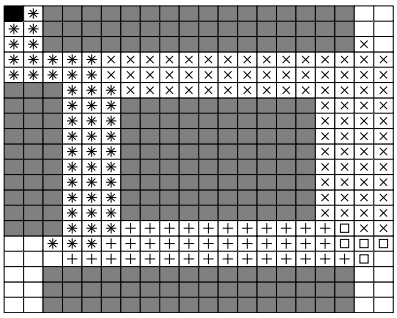
The calculation approach of $\hat{\pi}^s$ can be seen in [25].

We apply our compression method to the traffic model and the definition of parametric fuzzy sets in heuristic search is the same as in Subsection 4.2 with $|\mathcal{S}^c|=4$. For comparison, we also compress $\mathcal{H}^0$ using the spectral clustering and entropy methods proposed in [25, 26]. These two methods are in fact MC compression methods, which



(a) Spectral method

(b) Entropy method



(c) Aggregation derived from $\hat{R}^c$

FIG. 4.9. *Depictions of state aggregations found by the three different compression methods. Equal signs refer to assignment to the same aggregate state. The empty squares are unreachable from the start area and thus not part of the compressed state space.*

reduce the size of $\mathcal{H}^0$ through state aggregation only according to A^0. Fig. 4.9 shows the state aggregation results obtained by the three methods, where $\hat{R}^c = \left[\hat{R}^c_{ij}\right]$ and

$$\hat{R}^c_{ij} = 1_{\left\{i = \arg\max\limits_k R^c_{kj}\right\}} \tag{4.23}$$

In addition, we generate 1000 trajectories of the agent, and use the full model and compressed models to get estimates of c_t. Table 4.1 summarizes the estimation performance of the models, where

$$\mathrm{ER} = \frac{1}{L-1}\sum_{t=1}^{L-1}\left(1 - 1_{\{\hat{c}_t = c_t\}}\right) \tag{4.24}$$

is the error rate. Fig. 4.10 plots the trajectory generated from a single run and Fig. 4.11 displays corresponding the posterior probabilities of the corridors at each time, where $\delta_i^C(t) = 1_{\{c_t = i\}}$. As observed from the table and figures, in comparison with the MC compression based methods, our approach takes into account the influence of spatial distribution of sensors and corridors when compressing, and estimates the value of c_t more accurately.

5. Conclusions. We have presented a computationally efficient approach to optimally compress HMMs. Optimality is defined in terms of the similarity between the outputs of the original and the compressed model. The optimization procedure

TABLE 4.1

Estimation results. This table shows the mean and variance of the ER *calculated over* 1000 *independent simulations.*

	full model	our method	spectral method	entropy method
mean of ER	0.1489	0.2121	0.5276	0.3581
variance of ER	0.0144	0.0292	0.0633	0.0403

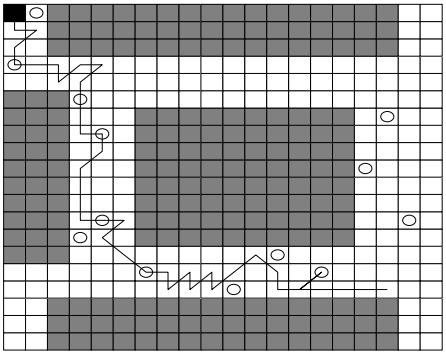
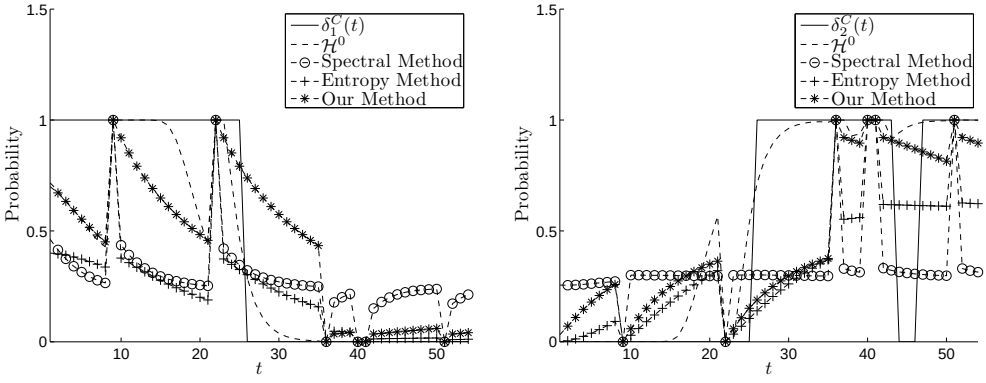
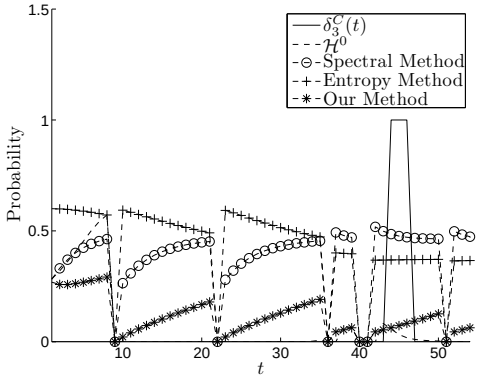


FIG. 4.10. *A sample trajectory of the agent*



(a) $c_t = 1$

(b) $c_t = 2$



(c) $c_t = 3$

FIG. 4.11. *Estimated* $p\left(c_t|o'_{1:t}\right)$ *for different* c_t.

does not evaluate these outputs explicitly, but rather minimizes a function which is an upper bound to the distance between the output distributions. This makes the presented approach efficient and permits applications that require large-scale stochastic models. The method is straightforwardly applicable to normal MC compression if the output probabilities are taken to be unit vectors.

The main result of the method is the fuzzy association matrix R which assigns the states in the large original state space to the small state space in a probabilistic way and is therefore a more general result compared to stochastic compression methods that assign states in a crisp way (*i.e.* lumping). Based on R a lumping can subsequently be performed by assigning each state of the original state space to the state of the compressed state space it belongs to most. Such a procedure will reduce the quality of the compressed model compared to the fuzzy compression, but is more easily interpreted and its error can be evaluated a posteriori by comparing the output distributions of the original and the lumped model. This is most easily done by evaluating the upper bound suggested in this paper.

Stochastic compression of HMMs will be useful for a wide variety of applications in different fields of the quantitative sciences. The following are examples of some possible applications:

1. **MC compression**: The method is straightforwardly applicable to normal MC compression if the output probabilities are taken to be unit vectors.
2. **Compression of molecular dynamics in terms of experimental observables**: The HMM compression method is an extension of MC compression in the sense that the property of observations are accounted for. As in the toy example of Subsection 4.2, two metastable states can be merged when they have similar observation probabilities. Therefore, our method can be applied to Markov models of molecular dynamics derived from simulations [28] with an observation probability distribution that mimics the observable of an experiment. In this way, effective models of MD as they appear to the experimental observer, can be computed. Subsequently, the essential stationary and dynamical features of the output sequence observed in the experiment can be interpreted in terms of the simulated structures that are encoded in the uncompressed state space of the original HMM.
3. **Reduced order estimation in signal processing**: This problem arises in many engineering fields, especially positioning, navigation and tracking. In these areas, the large size of original target movement model will cause difficulty for real-time estimation. The example of Subsection 4.3 shows that the proposed compression method has an application potential for reduced order estimator design.

But for the practical application of our methods, two problems still require investigation in the future. First, the HMMs are inferred from statistical data in many practical cases, so the influence of the original model error on compression result requires further study. Second, practical large-scale problems need a more efficient algorithm, which will be designed based on further analyzing the objective function of compressed model optimization.

Appendix A. Projected gradient algorithm.

Here we describe the PGA used in this paper so that it is self-contained (see [6, 42] for more details).

The optimization problem (3.26) can be formulated as

$$\begin{array}{l} \min f(x) \\ \text{subject to:} \\ \quad x \in \Xi \end{array} \tag{A.1}$$

where the vector $x \in \mathbb{R}^{N^c(N^c+M+N)}$ consists of elements of A, B, R, and Ξ denotes the feasible set defined by the linear constraints in (3.26). The PGA is similar to the gradient descent algorithms for unconstrained optimization except that the PGA utilize the projection $\mathcal{P}_\Xi$,

$$\mathcal{P}_\Xi(x) = \arg\min_{y \in \Xi} \|y - x\|_2 \tag{A.2}$$

to make the solution feasible at each iteration.

The algorithm with initial solution $x^{(0)}$ can be stated as follows:

Let the numbers $\mu_{PG} \in (0,1)$, $\gamma_{PG} > 0$ and a small enough number $\epsilon_{PG} > 0$ be given.

Step 1 Let $\alpha_0 = 1$ and $k = 1$.

Step 2 Compute the least nonnegative integer l such that $\alpha_k = 4^{-l} \max\{2\alpha_{k-1}, \gamma_{PG}\}$ satisfies

$$f\left(x^{(k)}(\alpha_k)\right) \leq f\left(x^{(k)}\right) + \mu_{PG}\left(x^{(k)}(\alpha_k) - x^{(k)}\right)^{\mathrm{T}} \nabla f\left(x^{(k)}\right)$$

where $x^{(k)}(\alpha_k) = \mathcal{P}_\Xi\left(x^{(k)} - \alpha_k \nabla f\left(x^{(k)}\right)\right)$. Set $x^{(k+1)} = x^{(k)}(\alpha_k)$.

Step 3 Terminate if $\left\|x^{(k+1)} - x^{(k)}\right\|_2 \leq \epsilon_{PG}$, otherwise let $k := k+1$ and go to Step 2.

Appendix B. Estimation of distribution algorithm.

For an optimization problem $\max_{x \in \mathbb{R}^n} f(x)$, the EDA can be summarized by the following algorithm:

Step 1 Randomly generate P_p feasible solutions $\mathcal{D}_0 = \left\{x^{1,0}, \ldots, x^{P_p,0}\right\}$. Let $k = 1$.

Step 2 Select P_{se} best solutions $\mathcal{D}_{k-1}^S = \left\{y^{1,k}, \ldots, y^{P_s,k}\right\}$ from $\mathcal{D}_{k-1}$.

Step 3 Estimate an n-dimensional probability density function (pdf) $p^E(x)$ based on $\mathcal{D}_{k-1}^S$.

Step 4 Sample P_p new solutions $\mathcal{D}_k = \left\{x^{1,k}, \ldots, x^{P_p,k}\right\}$ from $p^E(x)$.

Step 5 Return the best solution found so far if $k = K_{\max}$, otherwise let $k := k+1$ and go to Step 2.

The main issue of implementing the EDA is how to estimate the accurate distribution that can capture the structure of the selected solutions in Step 3. Here we assume that $p^E(x)$ can be factorized according to

$$p^E(x) = \prod_{i=1}^{n} p_{\mathcal{N}}\left(x_i | \nu_i, \sigma_i^2\right)$$

where $p_{\mathcal{N}}\left(\cdot | \nu, \sigma^2\right)$ denotes the pdf of the normal distribution with mean ν and variance σ^2, and parameters $\left\{\nu_i, \sigma_i^2\right\}$ can be estimated by the maximum likelihood method [7].

Appendix C. Agent movement model.

Let

$$N_b(x,y) = \{(x',y') \,|\, |x'-x| + |y'-y| \leq 1\} \tag{C.1}$$

be the neighbor set of grid (x, y), and $G = (\mathcal{V}, \mathcal{E})$ be a graph model with vertex set

$$\mathcal{V} = \{(x, y) \,|\, (x, y) \text{ is not an obstacle grid}\} \tag{C.2}$$

and edge set

$$\mathcal{E} = \{((x, y), (x', y')) \,|\, (x', y') \in N_b(x, y) \setminus \{(x, y)\}, (x', y'), (x, y) \in \mathcal{V}\} \tag{C.3}$$

From time t to $t + 1$,

$$(x_{t+1}, y_{t+1}) = \begin{cases} (x_t, y_t), & u_t < 0.2 \\ (x'_{t+1}, y'_{t+1}), & u_t \geq 0.8 \end{cases} \tag{C.4}$$

where $u_t \overset{\text{iid}}{\sim} \mathcal{U}_{[0,1]}$, *i.e.*, the agent will stay in the same grid with probability 0.2. The probability model of (x'_{t+1}, y'_{t+1}) consists of the following two parts:

- Movement toward the exit:

  $$(x'_t, y'_t) \,|\, (x_t, y_t) \sim \mathcal{U}_{\underset{(x,y)\in N_b(x_t,y_t)}{\arg\min} V(x,y)}$$

  where $V(x, y)$ is the distance of the shortest path from (x, y) to the exit in the graph G under the assumption that the weight of each edge is 1, and can be calculated by the Dijkstra algorithm [8].
- Noise:

  $$(x'_{t+1}, y'_{t+1}) \,|\, (x'_t, y'_t, x_t, y_t) \sim p\left(x'_{t+1}, y'_{t+1} | x'_t, y'_t, x_t, y_t\right)$$

  where

  $$p\left(x'_{t+1}, y'_{t+1} | x'_t, y'_t, x_t, y_t\right) \propto \begin{cases} 2, & (x'_{t+1}, y'_{t+1}) = (x'_t, y'_t) \\ 1, & (x'_{t+1}, y'_{t+1}) \in N_o \setminus \{(x'_t, y'_t)\} \\ 0, & \text{otherwise} \end{cases}$$

  and

  $$\begin{aligned} N_o = \{(x, y) \,|\, &(x - x'_t)(x'_t - x_t) + (y - y'_t)(y'_t - y_t) = 0\} \\ &\cap N_b(x'_t, y'_t) \cap \mathcal{V} \end{aligned}$$

In other words, the agent tries to move along the direct path towards, but is disturbed by some noise which causes it to make random side-steps.

REFERENCES

[1] R. W. Aldhaheri and H. K. Khalil, *Aggregation and optimal control of nearly completely decomposable Markov chains*, in Proceedings of the 28th Conference on Decision and Control, Tampa, Florida US, December 1989, pp. 1277–1282.

[2] ———, *Aggregation of the policy iteration method for nearly completely decomposable Markov chains*, IEEE Trans. Auto. Control, 36 (1991), pp. 178–187.

[3] O. Becker and M. Karplus, *The topology of multidimensional potential energy surfaces: Theory and application to peptide structure and kinetics*, J. Chem. Phys., 106 (1997), pp. 1495–1517.

[4] J. Bezdek, R. Ehrlich, et al., *FCM: The fuzzy c-means clustering algorithm*, Computers and Geosciences, 10 (1984), pp. 191–203.

[5] V. Blue, M. Embrechts, and J. Adler, *Cellular automata modeling of pedestrian movements*, in Proceedings of the 1997 IEEE International Conference on Systems, Man, and Cybernetics, vol. 3, 1997, pp. 2320–2323.

[6] P. Calamai and J. Moré, *Projected gradient methods for linearly constrained problems*, Mathematical Programming, 39 (1987), pp. 93–116.
[7] G. Casella, R. Berger, and R. Berger, *Statistical inference*, Duxbury Pacific Grove, CA, 2002.
[8] T. Cormen, *Introduction to algorithms*, MIT press, 2001.
[9] T. Dayar and W. Stewart, *Quasi lumpability, lower-bounding coupling matrices, and nearly completely decomposable Markov chains*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 482–498.
[10] P. Deuflhard and M. Weber, *Robust Perron cluster analysis in conformation dynamics*, Linear Algebra Appl., 398 (2005), pp. 161–184.
[11] S. Dey and I. Mareels, *Reduced-complexity estimation for large-scale hidden Markov models*, IEEE Trans. Signal Process., 52 (2004), pp. 1242–1249.
[12] S. Eddy, *Hidden Markov models*, Curr. Opin. Struct. Biol., 6 (1996), pp. 361–365.
[13] L. Finesso, A. Grassi, and P. Spreij, *Approximation of stationary processes by hidden Markov models*, Arxiv preprint math/0606591, (2006).
[14] I. Gerontidis and S. Kontakos, *Markov chain lumpability on fuzzy partitions*, in Proceedings of the IEEE International Fuzzy Systems Conference, 2007, pp. 1–6.
[15] A. L. Gibbs and F. E. Su, *On choosing and bounding probability metrics*, Intl. Statist. Rev., 70 (2002), pp. 419–435.
[16] I. Horenko, S. Dolaptchiev, A. Eliseev, I. Mokhov, and R. Klein, *Metastable decomposition of high-dimensional meteorological data with gaps*, J. Atm. Sci., 65 (2008), pp. 3479–3496.
[17] H. Jaeger, *Observable operator processes and conditioned continuation representations*, Neural Comput., 12 (2000), pp. 1371–1398.
[18] J. Kemeny and J. Snell, *Finite Markov chains*, Springer, 1976.
[19] G. Kotsalis, *Model Reduction for Hidden Markov Models*, PhD thesis, Massachusetts Institute of Technology, 2007.
[20] G. Kotsalis, A. Megretski, and M. Dahleh, *A model reduction algorithm for hidden Markov models*, in Proceedings of the IEEE Conference on Decision and Control, 2006, pp. 3424–3429.
[21] V. Kriihnamurthy, *Adaptive estimation of hidden nearly completely decomposable Markov chains*, in Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, 1994, pp. 337–340.
[22] V. Krishnamurthy, *Adaptive estimation of hidden nearly completely decomposable Markov chains with applications in blind equalization*, Int. J. Adap. and Signal Process., 8 (1994), pp. 237–260.
[23] P. Larranaga, R. Etxeberria, J. Lozano, and J. Pena, *Optimization by learning and simulation of Bayesian and Gaussian networks*, Tech. Report EHU-KZAAIK-4/99, University of the Basque Country, December 1999.
[24] M. Mohle, *General Applied Probability-Total variation distances and rates of convergence for ancestral coalescent processes in exchangeable population models*, Adv. in Appl. Probab., 32 (2000), pp. 983–993.
[25] J. Niedbalski, *Full and reduced order estimation of traffic dynamics using markov chains*, master's thesis, University of Illinois at Urbana-Champaign, 2007.
[26] J. Niedbalski, K. Deng, P. Mehta, and S. Meyn, *Model reduction for reduced order estimation in traffic models*, in Proceedings of American Control Conference, 2008, pp. 914–919.
[27] F. Noé and S. Fischer, *Transition networks for modeling the kinetics of conformational change in macromolecules*, Curr. Opin. Struct. Biol., 18 (2008), pp. 154–162.
[28] F. Noé, C. Schütte, E. Vanden-Eijnden, L. Reich, and T. Weikl, *Constructing the full ensemble of equilibrium folding pathways from short off-equilibrium simulations*, PNAS, (2009). in Press.
[29] J. Norris, *Markov chains*, Cambridge University Press, 1998.
[30] S. Ozkan, K. Dill, and I. Bahar, *Computing the transition state populations in simple protein models*, Biopolymers, 68 (2003), pp. 35–46.
[31] B. Peng, *Convergence, Rank Reduction and Bounds for the Stationary Analysis of Markov Chains*, PhD thesis, North Carolina State University, 2004.
[32] J. Peng and Z. Lin, *A non-interior continuation method for generalized linear complementarity problems*, Mathematical Programming, 86 (1999), pp. 533–563.
[33] L. Puztig, D. Becherer, and I. Horenko, *Optimization of futures portfolio utilizing numerical market phase-detection*. submitted to *SIAM J. Fin. Math.*
[34] L. Rabiner, *A tutorial on hidden Markov models and selected applications inspeech recognition*, Proc. IEEE., 77 (1989), pp. 257–286.

[35] L. Rabiner and B. Juang, *An introduction to hidden Markov models*, IEEE ASSP Magazine, 3 (1986), pp. 4–16.
[36] W. Rungsarityotin, R. Krause, A. Schödl, and A. Schliep, *Identifying protein complexes directly from high-throughput TAP data with Markov random fields*, BMC Bioinformatics, 8 (2007), p. 482.
[37] A. Schliep, A. Schonhuth, and C. Steinhoff, *Using hidden Markov models to analyze gene expression time course data*, Bioinformatics, 19 (2003), pp. I255–I263.
[38] C. Schütte, A. Fischer, W. Huisinga, and P. Deuflhard, *A direct approach to conformational dynamics based on hybrid Monte Carlo*, J. Comp. Phys., 151 (1999), pp. 146–168.
[39] L. Shue and S. Dey, *Complexity reduction in fixed-lag smoothing for hidden Markov models*, IEEE Trans. Signal Process., 50 (2002), pp. 1124–1132.
[40] W. Spears, *The role of mutation and recombination in evolutionary algorithms*, PhD thesis, George Mason University, 1998.
[41] W. M. Spears, *A compression algorithm for probability transition matrices*, SIAM J. Matrix Anal. Appl., 20 (1998), pp. 60–77.
[42] W. Sun and Y. Yuan, *Optimization theory and methods: nonlinear programming*, Springer, 2006.
[43] D. Wales, *Energy landscapes*, Springer, 2003.
[44] M. Weber, *Improved perron cluster analysis*, Tech. Report ZIB-Report 03-04, Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB), Germany, April 2003.
[45] L. White, R. Mahony, and G. Brushe, *Lumpable hidden Markov models-model reduction and reduced complexity filtering*, IEEE Trans. Automat. Control, 45 (2000), pp. 2297–2306.
[46] L. Zadeh, *Probability measures of fuzzy events*, J. Math. Anal. Appl., 23 (1968), pp. 421–427.
[47] J. Zhu, *Mining Web Site Link Structures for Adaptive Web Site Navigation and Search*, PhD thesis, University of Ulster, 2003.